Variant calling with GATK

Variant calling: Sophisticated

The GATK variant pipeline is the current "best practices" model for variant calling in human genome and exome data. Exercises will be shown to illustrate the various steps, but we don't have time to cover all the steps, tools, and methodologies used.

A few take-home points:

- A simple problem (finding SNPs) done at large-scale (human genome) introduces significant false positives which can take a lot of time and work to deal with
- Even if your data isn't human-sourced, you may find valuable ideas by looking over more mature pipelines from large genome projects
- There are many, many people constantly evolving computational methods and pipelines like GATK, so chose carefully before you start, and recognize that switching mid-stream may be costly
- Amongst all this sophistication, keep in mind the boldface caveat posted on the GATK site: **Care should be taken by the analyst running our tools to understand what each parameter does and to evaluate which value best fits his/her data** - use common sense, and sanity check your results frequently!

We've taken the liberty of simply incorporating the GATK wiki site by iframe into this course.

Scott/Dhivya's script for actually running v3 GATK at TACC on Lonestar:

```
#!/bin/bash

echo "exome_pipeline_ill.bash r1fastqfile r2fastqfile refindex reference dbsnp outprefix"
date
# echo "expected f3_f5prefix.F5.csfasta f3_f5prefix.F5_QV.qual f3_f5prefix.F5.csfasta f3_f5prefix.F5_QV.qual "

#read f3readfile f3qualfile f5readfile f5qualfile refindex reference dbsnp outprefix

f3fastqfile=$1
f5fastqfile=$2
refindex=$3
reference=$4
dbsnp=$5
outprefix=$6

f3fastqfile_prefix=$outprefix."r1"
f5fastqfile_prefix=$outprefix."r2"
# f3fastqfile=$f3fastqfile_prefix."single.fastq"
# f5fastqfile=$f5fastqfile_prefix."single.fastq"

echo "exome_pipeline_ill.bash $f3fastqfile $f5fastqfile $refindex $reference $dbsnp $outprefix"

echo "bwa aln $refindex $f3fastqfile > $f3fastqfile_prefix.sai 2>$f3fastqfile_prefix.bwa.log "
# bwa aln $refindex $f3fastqfile > $f3fastqfile_prefix.sai 2>$f3fastqfile_prefix.bwa.log
echo "bwa aln for $f3fastqfile done"
date
bwa aln $refindex $f5fastqfile > $f5fastqfile_prefix.sai 2>$f5fastqfile_prefix.bwa.log
echo "bwa aln for $f5fastqfile done"
date

#bwa sampe and awk to get only mapped data, also with read group (RG) info
bwa sampe -A -a 600 -r '@RG\tID:noID\tPL:ILLUMINA\tLB:noLB\tSM:bar' $refindex $f3fastqfile_prefix.sai
$f5fastqfile_prefix.sai $f3fastqfile $f5fastqfile > $outprefix.sam 2> $outprefix.sampe.log
echo "bwa sampe for $f3fastqfile_prefix done"
date

#sam to bam conversion
#doesn't work
#java -Xmx4g -Djava.io.tmpdir=/tmp -jar /home/daras/picard-tools-1/picard-tools-1.53/SortSam.jar SO=coordinate
INPUT=$outprefix.sam OUTPUT=$outprefix.bam VALIDATION_STRINGENCY=LENIENT

#mark PCR duplicates
#doesn't work
#java -Xmx4g -Djava.io.tmpdir=/tmp -jar /home/daras/picard-tools-1/picard-tools-1.53/MarkDuplicates.jar
INPUT=$outprefix.bam OUTPUT=$outprefix.marked.bam METRICS_FILE=metrics CREATE_INDEX=true
VALIDATION_STRINGENCY=LENIENT
```

```
#sam to bam conversion using samtools
samtools view -b -S $outprefix.sam > $outprefix.bam
echo "samtools view for $outprefix.sam done"
date
rm $outprefix.sam
samtools sort $outprefix.bam $outprefix.sorted
date
mv $outprefix.sorted.bam $outprefix.bam
echo "samtools sort for $outprefix.bam done"
date
samtools index $outprefix.bam
echo "samtools index $outprefix.bam done"
date

#local realignment around indels
java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -T
RealignerTargetCreator -R $reference -o $outprefix.bam.list -I $outprefix.bam 2>$outprefix.indel.log
java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -I
$outprefix.bam -R $reference -T IndelRealigner -targetIntervals $outprefix.bam.list -o $outprefix.realigned.bam
2>$outprefix.indel2.log
date

#fix mate info   -MAY REMOVE
# java -d64 -Xms512m -Xmx4g -Djava.io.tmpdir=/tmp -jar /opt/picard-tools-1/picard-tools-1.53/FixMateInformation.
jar INPUT=$outprefix.realigned.bam OUTPUT=$outprefix.realigned.fixed.bam SO=coordinate
VALIDATION_STRINGENCY=LENIENT CREATE_INDEX=true


#quality score recalibration
java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -l INFO -R
$reference -knownSites $dbsnp -I $outprefix.realigned.bam -T CountCovariates -cov ReadGroupCovariate -cov
QualityScoreCovariate -cov CycleCovariate -cov DinucCovariate -recalFile $outprefix.recal_data.csv 2>$outprefix.
recal.log
date

java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -l INFO -R
$reference -I $outprefix.realigned.bam -T TableRecalibration -o $outprefix.realigned.recal.bam -recalFile
$outprefix.recal_data.csv 2>$outprefix.recal2.log

#Produce SNP calls

java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -glm BOTH -
R $reference -T UnifiedGenotyper -I $outprefix.realigned.recal.bam --dbsnp $dbsnp -o $outprefix.snps.vcf -
metrics snps.metrics -stand_call_conf 50.0 -stand_emit_conf 10.0 -dcov 1000 -A DepthOfCoverage -A AlleleBalance
date

#filter SNPs (according to seqanswers exome guide)
java -d64 -Xms512m -Xmx4g -jar /work/01866/phr254/gshare/Tools_And_Programs/bin/GenomeAnalysisTK.jar -R
$reference -T VariantFiltration -B:variant,VCF snp.vcf.recalibrated -o $outprefix.snp.filtered.vcf --
clusterWindowSize 10 --filterExpression "MQ0 >= 4 && ((MQ0 / (1.0 * DP)) > 0.1)" --filterName
"HARD_TO_VALIDATE" --filterExpression "DP < 5 " --filterName "LowCoverage" --filterExpression "QUAL < 30.0 " --
filterName "VeryLowQual" --filterExpression "QUAL > 30.0 && QUAL < 50.0 " --filterName "LowQual" --
filterExpression "QD < 1.5 " --filterName "LowQD" --filterExpression "SB > -10.0 " --filterName "StrandBias"
date
```

NOTE: If you are looking to setup GATK yourself, here are some tips:

See the best practices section:

http://www.broadinstitute.org/gatk/guide/best-practices

Download the resource bundle:

http://www.broadinstitute.org/gatk/guide/article?id=1213

If you are looking to run GATK at TACC on Lonestar or Stampede with data from human samples, here are some tips:

GATK resource bundles are kept in the BioITeam corral directory here:

`/corral-repl/utexas/BioITeam/ref_database/GATK/`

You will need to use `module spider GATK` to figure out which versions of GATK are currently installed, and then use the appropriate resource bundles.